Gupta K.M., Aha D.W., & Moore P.G. (2006). Rough set feature selection algoritms for textual case-based classification. To appear in M.G.. Göker M.G. & T.R. Roth-Berghofer (Eds.) Proceedings of Eigth European Conference on Case-Based Reasoning, ECCBR-06, Ölündeniz, Turkey: Springer.

Rough Set Feature Selection Algorithms for Textual Case-Based Classification

Kalyan Moy Gupta[1], David W. Aha[2], and Philip Moore[3]

[1]Knexus Research Corp.; Springfield, VA 22153; USA
[2]Naval Research Laboratory (Code 5515); Washington, DC 20375; USA
[3]ITT Industries; AES Division; Alexandria, VA 22303; USA
firstname.lastname@nrl.navy.mil

Abstract. Feature selection algorithms can reduce the high dimensionality of textual cases and increase case-based task performance. However, conventional algorithms (e.g., information gain) are computationally expensive. We previously showed that, on one dataset, a rough set feature selection algorithm can reduce computational complexity without sacrificing task performance. Here we test the generality of our findings on additional feature selection algorithms, add one data set, and improve our empirical methodology. We observed that features of textual cases vary in their contribution to task performance based on their part-of-speech, and adapted the algorithms to include a part-of-speech bias as background knowledge. Our evaluation shows that injecting this bias significantly increases task performance for rough set algorithms, and that one of these attained significantly higher classification accuracies than information gain. We also confirmed that, under some conditions, randomized training partitions can dramatically reduce training times for rough set algorithms without compromising task performance.

1 Introduction

Textual case-based reasoning (TCBR) is a case-based reasoning (CBR) subfield concerned with the use of textual knowledge sources (Weber *et al*., 2005). TCBR systems differ in the degree to which their text content is used; some are *weakly textual* CBR while others are *strongly* textual CBR, meaning that textual information is the focus of reasoning (Wilson & Bradshaw, 2000). Applications such as email categorization, news categorization, and spam filtering require the use of strongly textual CBR methodologies. Most of these systems use a bag-of-words or term-based representation for cases (e.g., Wiratunga *et al.,* 2004; Delany *et al*., 2005), which can be problematic for textual case bases that have thousands of features. For example, this huge dimensionality could reduce accuracies on classification tasks and/or result in large computational costs.

A variety of feature selection algorithms can be used to address this issue. For example, these include conventional algorithms such as document frequency, information gain, and mutual information (Yang & Pederson, 1997). Wiratunga *et al.* (2004) extended these algorithms to include boosting and feature generalization with considerable success. However, some of these conventional algorithms have high computational complexity, which can be a problem when a TCBR system is applied to dynamic decision environments that require frequent case base maintenance.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2006		**00-00-2006 to 00-00-2006**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Rough Set Feature Selection Algorithms for Textual Case-Based Classification	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Knexus Research Corp,9120 Beachway Lane,Springfield,VA,22153**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Feature selection algorithms can reduce the high dimensionality of textual cases and increase case-based task performance. However, conventional algorithms (e.g., information gain) are computationally expensive. We previously showed that, on one dataset, a rough set feature selection algorithm can reduce computational complexity without sacrificing task performance. Here we test the generality of our findings on additional feature selection algorithms, add one data set, and improve our empirical methodology. We observed that features of textual cases vary in their contribution to task performance based on their part-of-speech, and adapted the algorithms to include a part-of-speech bias as background knowledge. Our evaluation shows that injecting this bias significantly increases task performance for rough set algorithms, and that one of these attained significantly higher classification accuracies than information gain. We also confirmed that, under some conditions, randomized training partitions can dramatically reduce training times for rough set algorithms without compromising task performance.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**16**	

Feature selection algorithms based on rough set theory (RST) rather than conventional algorithms can potentially alleviate this high computational complexity and also increase the task performance of TCBR systems. RST is a relatively novel approach for decision making with incomplete information (Pawlak, 1991). Feature selection algorithms motivated by RST have been applied with much success in non-textual CBR systems (e.g., Pal & Shiu, 2004). Recently, these algorithms have been applied to textual data sets. For example, Chouchoulas and Shen (2001) applied a rough set algorithm called QuickReduct to select features for an email categorization task. Also, we examined a rough set feature selection algorithm, called *Johnson's reduct*, to a multi-class classification problem (Gupta *et al.*, 2005). We empirically demonstrated that this algorithm, for one data set, was an order of magnitude faster than information gain and yet provided comparable classification performance. We also introduced a methodology that randomly partitions a training set, and selects and merges features from each partition. This *randomized training partitions* procedure can dramatically reduce feature selection time. We showed that its combination with Johnson's reduct was effective.

In this paper, we extend our earlier work on feature selection for TCBR classification tasks by exploring additional rough set algorithms. In particular, we introduce a variant of Li *et al.*'s (2006) relative dependency metric, called the *marginal relative dependency metric*, and explore its effectiveness with randomized training partitions. In addition, we introduce the notion of *part-of-speech bias* in textual case bases. This is based on our observation that textual features with different parts of speech may inherently differ in their ability to contribute to reasoning. For example, noun features may contribute more than verb features, as described in Section 3.4. Adapting rough set and conventional feature selection algorithms to incorporate this bias could improve their performance. We empirically investigate these issues on two data sets.

The rest of this paper is organized as follows. Section 2 introduces RST and two of its derivative feature selection algorithms. We also include a description of randomized training partitions and introduce the notion of part-of-speech bias. We present an empirical evaluation of the feature selection algorithms and their interaction with randomized training partitions and part-of-speech bias in Section 3. We review related work on feature selection in Section 4 and conclude with a discussion of our plans for future research in Section 5.

2 Rough Set Theoretic Feature Selection

2.1 Building Blocks of Rough Set Theory

For the sake of clarity for this audience, we use established CBR terminology, such as *cases* and *features*, to present the elements of RST. RST is based on a formal description of an information system (Pawlak, 1991). An information system S is a tuple $S = \langle C, F, V \rangle$ where:

$C = \{c_1, c_2, \ldots, c_n\}$ denotes a non-empty, finite set of *cases*,
$F = \{f_1, f_2, \ldots, f_m\}$ denotes a non-empty, finite set of *features* (or *attributes*), and
$V = \{V_1, V_2, \ldots, V_m\}$ is the set of value domains for the features in F.

A decision table is a special case of an information system where we distinguish two kinds of features: (1) a class (or *decision*) feature f_d, and (2) the standard conditional features F_p, which are used to predict the class of a case. Therefore, $F = F_p \cup \{f_d\}$.

Table 1. A case base fragment for hiring decisions

Cases	f_1 = age	f_2 = experience	f_3= grades	f_d =hired
c_1 = Anna	21-30	none	good	yes
c_2 = Bill	21-30	none	good	no
c_3= Cathy	21-30	4-6	average	no
c_4 = Dave	31-40	1-3	excellent	yes
c_5 = Emma	31-40	4-6	good	yes
c_6 = Frank	31-40	4-6	good	yes

We will explain RST concepts using the trivial case base in Table 1, which pertains to making hiring decisions based on three features. Central to RST is the notion of *indiscernibility*. Examining the cases in Table 1, we see that cases c_1=Anna and c_2=Bill have identical values for all the features, and thus are *indiscernible* with respect to the three conditional features f_1, f_2, and f_3. More broadly, a set of cases C' is indiscernible with respect to a set of features $F' \subseteq F$ if the following is true:

$$IND(F',C)= \{ C' \subseteq C \mid \forall f \in F', \forall c_i, c_{j\,(i \neq j)} \in C' \; f(c_i) = f(c_j)\} \quad (1)$$

Thus, two cases are indiscernible with respect to features in F' if they have identical values for all the features in F'.

An indiscernibility relation is an equivalence relation that partitions the set of cases into equivalence classes. Each equivalence class contains a set of indiscernible cases for the given set of features F'. For example, given the hiring decision table:

$$IND(F', C) = \{\{ c_1 , c_2\}, \{ c_3 \}, \{ c_4 \}, \{ c_5 , c_6\}\}$$

where F'={*age, experience, grades*} and $C=\{c_1,c_2,c_3,c_4,c_5,c_6\}$. The equivalence class of a case c_i with respect to selected features F' is denoted by $[c_i]_{F'}$. Based on the equivalence classes, RST develops two kinds of set approximations. First, given sets $C' \subseteq C$ and $F' \subseteq F$, the *lower approximation* of C' with respect to F' is defined as:

$$lower(C, F', C') = \{c \in C \mid [c]_{F'} \subseteq C'\} \quad (2)$$

or the collection of cases whose equivalence classes are subsets of C'. Second, the *upper approximation* of C' with respect to F' is instead defined as:

$$upper(C, F', C') = \{c \in C \mid [c]_{F'} \cap C' \neq \varnothing\} \quad (3)$$

or the collection of cases whose equivalence classes have a non-empty intersection set with C'. A set of cases C' is *crisp* (or *definable*) if $lower(C, F', C') = upper(C, F', C')$, and is otherwise *rough*.

For example, in the hiring decision table, consider $C'_{\{hired=yes\}} = \{c_1, c_4, c_5, c_6\}$, then the lower and upper approximations of $C'_{\{hired=yes\}}$ with respect to F'={*age, experience, grades*} are:

$$lower(C, F', C'_{\{hired=yes\}}) = \{c_4, c_5, c_6\} \text{ and } upper(C, F', C'_{\{hired=yes\}}) = \{c_1, c_2, c_4, c_5, c_6\}$$

Case c_1 is not included in the lower approximation because its equivalence class $\{c_1,c_2\}$ is not a subset of $C'_{\{hired=yes\}}$. However, it is included in the upper approximation because its equivalence class has a non-empty intersection with $C'_{\{hired=yes\}}$.

Another important RST element is the notion of a set called the *positive region*. The positive region of a decision feature f_d with respect to $F' \subset F$ is defined as:

$$POS_{F'}(f_d,C) = \cup \{ lower(C, F', C') \mid C' \in IND(\{f_d\}, C)\} \quad (4)$$

or the collection of the F'-lower approximations corresponding to all the equivalence classes of f_d. For example, the positive region of f_d {hiring} with respect to F'={*age, experience, grades*}, where $lower(C, F', C'_{\{hired=no\}})=\{c_3\}$, is as follows:

$$POS_{F'}(f_d,C) = lower(C, F', C'_{\{hired=yes\}}) \cup lower(C, F', C'_{\{hired=no\}})=\{c_3, c_4, c_5, c_6\}$$

The positive region can be used to develop a measure of a feature's ability to contribute information for decision making. A feature $f \in F'$ makes no contribution or is *dispensable* if $POS_{F'}(f_d,C) = POS_{F'-\{fd\}}(f_d,C)$ and is *indispensable* otherwise. That is, removing the feature f_d from F' does not change the positive region of the decision feature. Therefore, ***features can be selected by checking whether they are indispensable*** with respect to a decision variable. The minimal set of features F', $F' \subset F$, is called a **reduct** if $POS_{F'}(f_d,C) = POS_F(f_d,C)$.

Often, an information system has more than one possible reduct. Generating a reduct of minimal length is a NP-hard problem. Therefore, in practice, algorithms have been developed to generate one "good" reduct. Next, we present our adaptations of two such algorithms: (1) Johnson's heuristic algorithm and (2) the marginal relative dependency algorithm.

2.2 Feature Selection with Johnson's Heuristic Algorithm

We adapted Johnson's (1974) heuristic to compute reducts as follows. It sequentially selects features by finding those that are most discernible for a given decision feature (see Figure 1). It computes a discernibility matrix M, where each cell $m_{i,j}$ of the matrix corresponding to cases c_i and c_j includes the conditional features in which the two cases' values differ. Formally, we define *strict discernibility* as:

$$m_{i,j} = \{\{ f \in F_p : f(c_i) \neq f(c_j)\} \text{ for } f_d(c_i) \neq f_d(c_j), \text{ and } \oslash \text{ otherwise } \} \quad (5)$$

JohnsonsReduct(F_p, f_d, C)
Input F_p: conditional features, f_d: decision feature, C: cases
Output R: Reduct $R \subseteq F_p$

```
1   R←⊘, F'←F_p
2   M← computeDiscernibilityMatrix(C, F', f_d)
3   do
4     f_h← selectHighestScoringFeature(M)
5     R ←R ∪ {f_h}
6     for (i=0 to |C|, j=i to |C|)
7       m_ij ← ⊘ if f_h ∈ m_ij
8     F'← F' – {f_h}
9   until m_ij = ⊘ ∀i, j
1   return R
```

0

Figure 1. Pseudocode for Johnson's heuristic algorithm

Given such a matrix M, for each feature, the algorithm counts the number of cells in which it appears. The feature f_h with the highest number of entries is selected for addition to the reduct R. Then all the entries m_{ij} that contain f_h are removed and the next best feature is selected. This procedure is repeated until M is empty.

The computational complexity of JohnsonsReduct is $O(VC^2)$, where V is the (typically large) vocabulary size and bounds the number of times the **do** loop is executed. However, this is a loose upper bound that is better approximated by $O(RC^2)$, where is $R<<V$. Comparing this complexity with the computational complexity of information gain, which is $O(MVC)$, where M is the number of classes, the complexity of JohnsonsReduct is lower because, typically, $RC<MV$. However, the worst case space complexity of JohnsonsReduct is $O(VC^2)$, which is significantly greater than Information Gain's space complexity of $O(VC)$.

In TCBR applications, each case may have only a small subset of features. *Strict* discernibility could be implemented as follows: $f(c_i) \neq f(c_j)$ if only one of the cases c_i or c_j contains the term denoted by the feature f. However, such an approach ignores the information contained in the variation of term frequencies (i.e., value) across cases. Hence, a graded or fuzzy notion of indiscernibility, instead of a strict notion, may be more effective (e.g., Skowron, 1995). We extend strict discernibility to *graded or fuzzy discernibility* using a similarity computation as follows. In Equation 5, we consider:

$$f(c_i) \neq f(c_j), \text{ when } sim(f(c_i), f(c_j)) < \tau_f \tag{6}$$

where $(0<\tau_f<1)$ is a user defined similarity threshold. We adapt a similarity measure for ordinal scales (Montazemi & Gupta, 1997) to compute the similarity between two non-zero frequency valued features as follows:

$$sim(f(c_i), f(c_j)) = \begin{cases} 1\text{-}abs((f(c_i)\text{-}f(c_j))/\psi.\sigma_f, \text{ when } abs((f(c_i)\text{-}f(c_j)) \leq \psi.\sigma_f \\ 0, \text{ otherwise} \end{cases} \tag{7}$$

where σ_f is the standard deviation of non-zero frequency values for feature f, and $\psi > 0$ is a user-defined parameter for adjusting similarity sensitivity. For example, for a feature f with $\sigma_f = 1.87$ and $\psi=1$,

$$sim(4, 5) = 1 - abs(4\text{-}5)/1.87*1 = 0.465$$

Similarly, the issue of class feature discernibility arises in TCBR for *multiclass* classification tasks in which more than one class can be assigned to a case. For example, topic assignment is a multi-class classification task. In Equation 5, we consider:

$$f_d(c_i) \neq f_d(c_j), \text{ when } sim(f_d(c_i), f_d(c_j)) < \tau_d \tag{8}$$

where $f_d(c_i)$ can be a set of values, $sim(f_d(c_i), f_d(c_j))$ yields the ratio of the intersection of its values to their union, and $0 < \tau_d < 1$ is a user defined similarity threshold.

2.3 Feature Selection using Marginal Relative Dependency

In Section 2.1, we described how an indiscernibility (or equivalence) relation partitions a case base C into equivalence classes with respect to a set of features F'.

Intuitively, with an increase in the number of features in F', we expect the number of equivalence classes to increase and each equivalence class to contain fewer cases. The *degree of relative dependency* of a set of features F' builds on this intuition. For a decision feature f_d and a set of features F', it is defined as (Li *et al*., 2006):

$$\delta_{F'}^{f_d} = \frac{|\Pi_{F'}(C)|}{|\Pi_{F' \cup f_d}(C)|} \tag{9}$$

where $\Pi_{F'}(C)$ is the set of equivalence classes generated over C with respect to features F' and $\Pi_{F' \cup f_d}(C)$ is the set of equivalence classes generated over C with respect to features $F' \cup \{f_d\}$. Clearly, the maximum value of $\delta_{F'}^{f_d}$ is 1. Based on this measure, we compute the marginal contribution of a feature f (i.e., marginal relative dependency), denoted by μ_f, as follows:

$$\mu_f = \delta_{F' \cup \{f\}}^{f_d} - \delta_{F'}^{f_d} \tag{10}$$

In addition to using μ_f as a metric for selecting features, it can also be used as a feature weight because $\sum_{f \in R} \mu_f = 1$, where R is a reduct.

Our variation on this reduct computation algorithm, called the *Marginal Relative Dependency* algorithm (MRD), is as follows (see Figure 2). At each iteration, it computes the marginal relative dependency of all the candidate features T, selects the feature f_m with the maximum marginal relative dependency, and adds it to the reduct R. The algorithm terminates when the relative dependency $\delta_R = \beta$, where β is a user defined parameter in the range ($0 < \beta < 1$). In a TCBR application, it is possible that beyond a certain point both μ_f and $\delta_{F'}^{f_d}$ may behave asymptotically. Therefore, β can be specified to terminate the feature selection process early.

MRD(F_p, f_d, C**)**
Input F_p: Conditional features, f_d: Decision feature, C: Cases, β: Threshold
Output R: Reduct $R \subseteq F_p$

```
1     R←⊘, F'←F_p, δ_R←0
3     do
4       <f_m, μ_m> ← selectMaximallyContributingFeatureAndValue(F',C)
5       R ←R ∪ {f_m}
6       F' ← F' - {f_m}
7       δ_R ← δ_R + μ_m
8     until δ_R = β
9     return R
```

Figure 2. Pseudocode for the Marginal Relative Dependency algorithm (MRD)

Like JohnsonsReduct, the determination of equivalence classes in MRD can be based on a strict or graded notion of discernibility. For the graded notion of discernibility we apply Equations 6, 7, and 8.

The worst case computational complexity of MRD is $O(RVC^2)$. For large textual case bases, this is an order of magnitude more complex than JohnsonsReduct and information gain. However, its worst case space complexity is only $O(VC)$.

2.4 Feature Selection with Random Training Set Partitions

The computational complexities of the feature selection algorithms discussed above depend on C, the number of training cases. The complexities of both RST approaches, JohnsonsReduct and MRD, are a function of the square of the number of training cases. Therefore, reducing the number of training cases that need to be considered at one time can dramatically reduce feature selection and training time. We can accomplish this by using randomized training partitions (RTP) (Gupta *et al.,* 2005), which is a procedure with the following steps:

1. Randomly create m equal-sized partitions of the training set.
2. From each partition, select features using a feature selection algorithm (e.g., JohnsonsReduct or MRD).
3. Define the final feature set as the union of features selected from each partition.

This approach could reduce the training time by a factor of m for the RST feature selection algorithms.

2.5 POS-Biaser: A Part-of-speech Bias Adjustment Method

In TCBR, words or terms are typically used as features. The linguistic attributes associated with such features (e.g., part-of-speech (POS), syntactic roles) could impact feature selection and TCBR task performance. For example, it is likely that noun features are generally more informative than verb features possibly because nouns are an *open class* of words, whereas verbs, adjectives, adverbs, prepositions, and pronouns are *closed classes* of words (Quirk *et al.*, 1985). Open word classes are frequently extended to include new words, whereas closed classes are rarely extended. Thus, a large percentage of terms in a typical vocabulary are nouns. However, each noun feature may occur in relatively fewer cases and has the potential to be more informative towards a decision. In contrast, verbs tend to occur more frequently across many cases. Also, there is considerable flexibility in the choice of verbs used to express the case content. This causes variability in verb expressions that could be inappropriately construed as informative (e.g., by information-theoretic measures) and as a result may be favored by feature selection algorithms. For example, this would adversely affect JohnsonsReduct, which relies on pair-wise case comparisons to construct a discernibility matrix. It is likely to select spurious verbs, as could MRD and information gain (IG) (Yang & Pederson, 1997).

One way to counter the effect of this inherent potential bias of textual case bases is to bias the feature selection algorithms accordingly. Thus, we introduce a simple methodology, called *POS-Biaser*, to use in combination with a feature selection algorithm. POS-Biaser assumes that part-of-speech tagging is performed during the case indexing process. This is feasible because part of speech taggers are publicly available (e.g., Brill, 1993). POS-Biaser uses a POS biasing factor ρ_{pos} for each POS

along with a feature selection metric to select features. For example, when $\rho_{noun} = 1.8$, $\rho_{verb} = 0.6$, $\rho_{adjective} = 1$, and $\rho_{adverb} = 0.3$, the feature selection algorithm's values for nouns are inflated to 1.8 times their original value, the values for verbs are deflated to 0.6 times their original value, and so on.

The POS-Biased JohnsonsReduct includes a modification to the step that executes *selectHighestScoringFeature*(*M*) (Figure 1, line 4), which computes the number of cell entries as the score of each feature (i.e., the feature selection metric). In particular, feature scores are now multiplied by their respective ρ_{pos} values. This would bias JohnsonsReduct to select more noun features than its unbiased version. Likewise, we accommodate a POS bias in MRD by similarly modifying the statement that executes *selectMaximallyContributingFeatureAndValue*(*F'*,*C*).

3 Evaluation

3.1 Claims and Empirical Methodology

We evaluated the feature selection algorithms described in this paper to explore the following hypotheses:

1. Rough set methods perform as well as or outperform information gain on our case-based classification tasks.
2. The performances of rough set feature selection algorithms are affected by the POS bias in textual case bases.
3. RTP is an effective way to dramatically reduce feature selection time without compromising case-based task performance.

We selected both a *single* and a *multi*-classification task to evaluate the utility of the feature selection and POS-biasing algorithms for a simple case-based classifier. Single classification involves assigning exactly one class label to a new text case, while multi-classification involves assigning one or more class labels. For example, sorting emails into a known set of folders is a single classification task and assigning one or more topic to news articles is a multi-classification task.

We selected tasks from two data sets, one for each type of classification task. The first data set is Reuters-21578 (Reuters, 2006); it contains news items and its multi-classification task concerns assigning topics to these items. The second data set is a subset of 20-News Groups (Lang, 2006); it contains news group emails and its single classification task concerns assigning a news group label to each of these emails. Due to the relatively high computational and space complexities of the algorithms being tested, we selected only the first ten news groups for evaluation in this data set; we call this 10-News Groups. Table 2 summarizes the characteristics of both data sets.

Table 2. A summary of the characteristics of the data sets used in the experiments

Characteristic	Reuters-21578	10-News Groups
Number of Cases	11,330 (with more than 0 topics)	10,013
Number of Classes	110	10
Num. Cases per class	103 (Avg.)	1001.3 (Avg.)
Num. Classes per Case	1.26 (Avg.), 1 (min.), 16 (max.)	1
Num. Words per case	137 (Avg.)	200.35 (Avg.)

We used two rough set feature selection algorithms (JohnsonsReduct (JR) and MRD) and one conventional feature selection algorithm, namely IG (Yang & Pederson, 1997). In the experiments, for a fair comparison, we ensured that all the algorithms selected the same number of features, and used JR to determine how many features to select. Finally, we also incorporated the POS bias in each feature selection algorithm, and refer to them as JRB, MRDB, and IGB, respectively.

Our feature generation algorithm performs tokenization, POS tagging, and morphotactic parsing to create POS-tagged terms as features. Morphotactic parsing is a more involved method than simple stemming; it reduces terms to their baseforms even across different POS (Gupta & Aha, 2004). For example, it reduces the noun "computer" to the verb "compute". Features with document frequency greater than two were considered for feature selection.

We applied a k-nearest neighbor classifier with the fuzzy feature similarity function described in Equation 7 to evaluate classification performance using the selected features. (We set k=5 based on feedback from our initial empirical studies.) All features were weighted equally to isolate the *selection* behaviors of the feature selection algorithms in our experiments. Multi-classification task performance was measured using *11-point average precision*, which is the average precision obtained at recall thresholds of (0%, 20%, …100%). The classifier assigns as many topics as needed until a given recall is achieved (Yang & Pederson, 1997). Performance on the single classification task was measured as classification accuracy. We also measured feature selection time (in seconds) for each algorithm.

We used a two-fold cross validation strategy to evaluate the algorithms. Two sets of two folds were randomly created. For RTP, all the algorithms were run with the same set of 10, 20, 30, and 40 randomized training partitions in each fold. We did not experiment without partitions due to the RTS algorithms' high computational and memory requirements.

3.2 Empirical Results

Results with the Reuters-21578 Data Set. The key results for the six algorithms (i.e., JR, IG, MRD, JRB, IGB, and MRDB) on this data set are shown in Figures 3-5. JR selected an average of 95.5, 118, 135, and 139.5 features for partitions of size 10, 20, 30, and 40, respectively. Increasing the number of RTP partitions increases the chance of selecting different features in different partitions, which increases the total number of unique features selected.

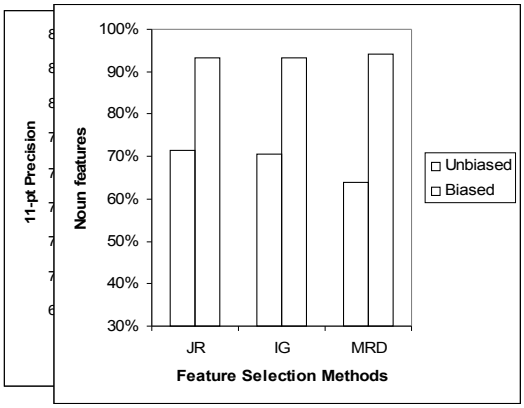

Figure 4. The effect of POS-bias on the number of noun features selected by the three algorithms for Reuters-21578 using 10 RTP partitions

We comparatively analyzed the algorithms' precision results using one-tailed paired student t-tests. Comparisons of the feature selection algorithms' unbiased versions show that JR significantly outperformed IG for every number of partitions tested (e.g., 76.72% vs. 70.17% at 10 partitions [p=.0006]), as did MRD (e.g., 79.21% vs. 75.86% at 40 partitions [p=.0018]). Therefore, *both the rough set feature selection methods significantly outperformed a conventional feature selection method*. In addition, MRD significantly outperformed JR at partitions of 30 and 40 (e.g., 79.20% vs. 77.83% at 40 partitions [p=.0003]), but the reverse was true for 10 partitions.

Comparing the POS-biased versions of the feature selection algorithms with their respective unbiased versions shows that JRB and IGB outperform JR and IG respectively at all RTP sizes. For example, at 30 partitions, JRB significantly outperforms JR (82.61% vs. 77.26% [p=.0007]) and IGB significantly outperforms IG (76.84% vs.74.79% [p=.0019]). However, MRDB significantly outperforms MRD only at 10 and 20 partitions; for 30 and 40 partitions there was no significant difference. Overall*, POS bias had a positive effect on all the feature selection algorithms, including IG*. It was most effective with JR, whose classification accuracy improved by 6.1% on average versus its unbiased version. Finally, *when adjusted for POS bias, JR recorded significantly higher precision results than the other feature selection algorithms we tested*.

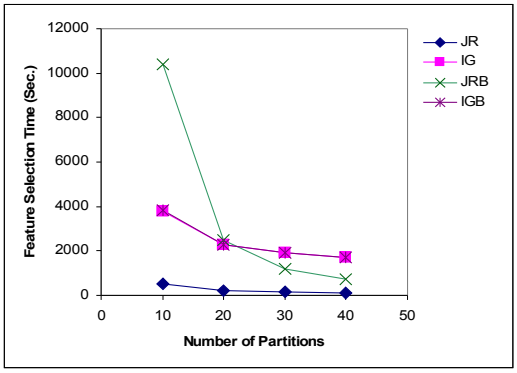

Figure 5. Feature selection times (Reuters-21578)

Figure 4 shows the effect of POS bias on the three feature selection algorithms for Reuters-21578 at 10 partitions. The proportion of noun features without bias were at comparable levels for JR and IG (each at 71%) and slightly

lower for MRD (64%). With this bias, the proportion of noun features increased to 93% for JR and IG and 94% for MRD. The increase in the proportion of noun features was comparable and consistent across the three algorithms, yet its effect on JR's precision performance was most substantial. Thus, we conclude that JR is most sensitive to POS bias.

Figure 5 shows the feature selection times for IG, JR, IGB, and JRB. *JR has the lowest feature selection time*. It decreased by 81.92% from 510 seconds at 10 partitions to 92 seconds at 40 partitions, without decreasing average precision, demonstrating that RTP is highly effective. Its biased version (JRB) has higher feature selection times (10,382 sec. at 10 to 738 sec. at 40 partitions) but achieves a similar decrease in feature selection time as the number of partitions increases. JRB's times are higher than JR's because POS bias significantly increases the reduct sizes. In contrast, IG and IGB have the same feature selection times. It reduces by 54% (3780 seconds to 1725 seconds) as the number of partitions is increased from 10 to 40. As expected, MRD has extremely long feature selection times (99,843 sec. at 10 partitions to 22,276 sec. at 40 partitions; not shown in Figure 5), and MRDB times are even longer. However, they both recorded a substantial drop in feature selection time as the number of partitions was increased. Therefore*, RTP is effective in reducing feature selection time on Reuters-21578 for the three algorithms we tested*.

Results with the 10-News Groups Data Set. As with the Reuters-21578 data set, we again used the number of features selected by JR as a baseline for the other algorithms. It selected an average of 123, 134.75, 141.25, & 153.5 features at 10, 20, 30, and 40 partitions, respectively.

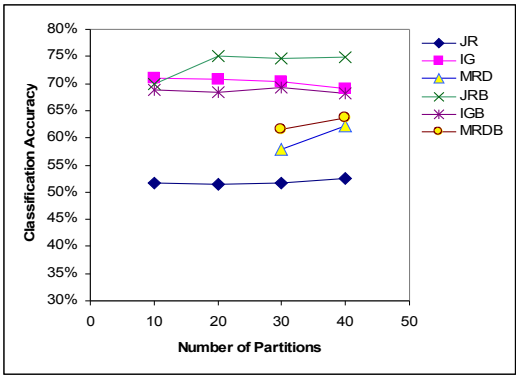

Figure 6. Classification accuracies (10-News Groups)

Comparison of the unbiased versions of the algorithms show that IG attains significantly higher accuracies than the others at all RTP levels on the 10-News Groups data (see Figure 6). For example, at 30 partitions, IG outperformed JR (70.31% vs. 51.74%, [p=.0005]) and MR (70.31% vs. 57.82%, p=.0005]). This contrasts with its comparatively poor precision performance on the Reuters-21578 data set.

Comparing the two rough set methodologies with each other reveals that MRD significantly outperformed JR at 30 and 40 partitions (e.g., 57.82 % vs. 51.74% at 30 partitions, [p=.022]). This finding is consistent with those on the Reuters data set. However, MRD's performance could not be objectively compared with JR at 10 and 20 partitions because it selected fewer features than JR at those partitions.

Comparing the algorithms' biased and unbiased versions show that JRB and MRDB attain significantly higher classification accuracies than JR and MRD, respectively. For example, JRB's average accuracy is significantly higher than JR's at 30 partitions (74.68% vs. 51.74%, [p=.0006]) and MRDB outperforms MRD (61.47% vs. 57.82%, [p=.022]). In contrast, IG was adversely affected by bias. That

is, IG performed slightly better than IGB (e.g., 70.31% vs. 69.36% at 30 partitions), although this difference was small and statistically insignificant. *Overall, JRB significantly outperformed the other algorithms at 20, 30, and 40 partitions.* For example, it attained significantly higher average classification accuracies than IG (74.7% vs. 70.3% at 30 partitions [p=.0018]).

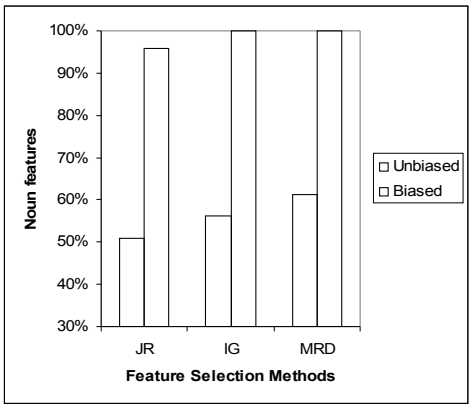

Figure 7. The effect of POS-bias on the number of noun features selected by the three algorithms for 10-News Groups using 30 RTP partitions

One possible reason could be that we used the same POS bias parameter settings for all the algorithms, but IG may require different settings. We gained additional insight into this by examining the effect of POS bias on the algorithms (see Figure 7). The unbiased versions of the algorithms selected different proportions of noun features; JR selected 51%, IG selected 55%, and MRD selected 61% at 30 partitions. Examining the biased versions shows that JRB selects 96%, while IGB and MRDB select 100%, indicating that the bias factors may be too strong for IG and MRD.

Analyses of the feature selection times shows that JR's times steadily decrease from 325 seconds at 10 partitions to 60 seconds at 40 partitions and is the lowest among all algorithms at 20-40 partitions (see Figure 8). Feature selection times for IG and IGB remain relatively constant (268 seconds, on average) across different partition sizes. In contrast, JRB's feature selection times decreased dramatically from 10 to 20 partitions, but increased from 30 to 40. This occurred because the decrease in the number of cases per partition is offset by larger increases in the reduct sizes, thereby leading to an overall increase in feature selection times. For the same reason MRD and MRDB's times steadily increase from 6291 seconds at 10 partitions to 10,134 seconds at 40 partitions (not shown in Figure 8). In general, MRD selects more features than JR and this is further amplified for higher numbers of partitions. *Thus, RTP significantly reduces feature selection times for only JR and JRB on the 10-News Group data set.*

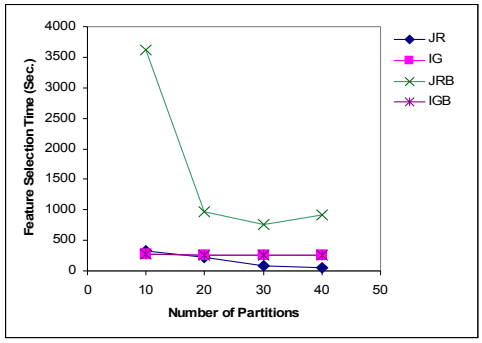

Figure 8. Feature selection times (10-News Groups)

Results Summary and Discussion. Given that one of the rough set methods, JR with suitable POS bias, outperformed IG on both the data sets, we partially accept our first hypothesis, which claims that rough set methods significantly outperform IG. We also confirmed our second hypothesis, which states that POS-bias has a positive effect on

RST feature selection algorithms. In particular, its effect on JR was substantial (6.1% increase in precision in Reuters-21578, and 41.78% increase in accuracy in 10-News Groups). Interestingly, the effect of POS-bias on IG was mixed: positive on Reuters-21578 and negative on 10-News Groups. We conjecture that the reasons for this mixed result are that the bias parameters for IG were too strong for the 10-News Groups set and that IG effectively counters the inherent POS bias when the number of cases per class is large (e.g., 1000 as opposed to 100).

We showed that the RTP was effective in dramatically reducing feature selection time for JR. However, the effect of RTP on MRD was mixed. It was positive on Reuters-21578 and negative on 10-News Groups. Therefore, we cannot fully confirm our third hypothesis that RTP is always effective in reducing training time for rough set methods. However, without RTP it would have been practically infeasible to run MRD and JR. We also observed that RTP has a positive effect on IG, although small compared to RST methods. This is because increasing the number of partitions reduces the effective vocabulary that IG must deal with and IG's computational complexity is linearly dependent on the vocabulary size.

4 Related Work

TCBR systems have been designed to support a variety of applications such as those involving legal reasoning (Brüninghaus & Ashley, 2003), spam filtering (Delany *et al.,* 2005), and news group classification (Wiratunga *et al.,* 2004). Typically, TCBR systems that use knowledge poor approaches (e.g., for email classification) tend to automatically generate features and operate on large data sets. For example, Delany *et al*. (2005) used IG to select features in a spam filtering task and Wiratunga *et al*., (2004) used IG to select features with boosted decision stumps. However, unlike us, they did not focus on reducing the computational complexity of their feature selection algorithms. Furthermore, high computational complexity was not a limiting factor because their binary classification task is not particularly demanding of information gain, especially given that their case bases were relatively small, containing only about 1000 cases. We instead investigate multi-classification and n-ary classification tasks involving thousands of cases, which require more attention to computational complexity. Despite these differences, our feature selection algorithms, randomized training partitions, and POS biasing can be effectively integrated with their approach.

Given a set of manually selected features, Brüninghaus & Ashley's (2003) TCBR system induces a set of classifiers that can automatically assign features to text documents. They used ID3 to induce these classifiers. If the number of features is large, its performance would degrade significantly. In such situations, our feature selection algorithms could significantly improve ID3's performance.

While RST-motivated feature selection algorithms have recently been applied to textual case bases on classification tasks, we are the first group to highlight complexity issues (Gupta *et al*., 2005). For example, Chouchoulas & Shen (2001) applied their QuickReduct method for email classification. While QuickReduct's complexity (Gupta *et al*., 2005) is high (i.e., the same as MRD), they did not address complexity because their data included only 1500 cases. Furthermore, they did not compare QuickReduct with any conventional feature selection algorithms, such as IG.

Li *et al.* (2006) developed a Fast Rough Set Feature Reduction algorithm. Unlike the RST algorithms we evaluated, it is not feasible to isolate the contributions of RST in their hybrid conventional/RST algorithm. In particular, they used IG to rank-order the features for selection and the relative dependency metric *only* to terminate feature selection. Finally, they did not compare the performance of their algorithm with conventional algorithms.

An *et al.* (2004) developed a rough set feature selection method called ELEM2 and applied it to web page classification. As with the other research groups, they did not address complexity issues and evaluated their algorithm on a relatively small set of 327 web pages. Moreover, they tested their algorithm only with the most frequently occurring 20, 30, and 40 keywords per category. Although this drastically reduces their data set's number of features, frequency-based keyword selection is not always competitive with other feature selection algorithms (Yang & Pederson, 1997).

In our previous research (Gupta *et al.,* 2005), we introduced RST motivated feature selection algorithms for a multi-class classification task. We also noted that the high computational complexity of feature selection algorithms are a limiting factor and introduced randomized training partitions to reduce training time. Finally, we showed that JohnsonsReduct performed comparably to IG on a single data set. In this paper, we extended JohnsonsReduct to work with multi-valued features and introduced the topic of fuzzy discernibility. In addition, we introduced MRD, a pure rough set version of Li *et al.*'s (2006) Fast Rough Set Reduction Approach. While this increases computational complexity, it is offset through the use of RTP. We also improved our evaluation methodology. For example, we eliminated variances due to differences in feature weighting by weighting all features equally, added a single classification task to improve the reliability of our conclusions, and used a two-fold cross validation methodology rather than random sampling. This has led us to qualitatively new results. For example, we found randomized training partitions to be effective for *both* rough set and conventional feature selection algorithms (for the Reuters-21758 data set), rather than only for the former.

Finally, we introduced the use of a POS-bias in textual case bases and described why it can impact feature selection. This explicit manipulation of bias appears to be novel; we are not aware of any prior research on using background knowledge of this type to assist TCBR systems on classification tasks. We showed that biasing feature selection algorithms can significantly increase classification accuracy of both conventional and RST-motivated feature selection algorithms, and that these increases are more substantial for the rough set algorithms.

5 Conclusion

Until recently, only conventional feature selection algorithms (e.g., IG and its extensions) had been applied to textual CBR with little concern for their computational complexity. In this paper, we rigorously investigated the potential of RST approaches to improve task performance and reduce feature selection times. We considered two RST algorithms: (1) JR with lower computational complexity than IG and (2) MRD with much higher computational complexity than IG. We evaluated the effect of RTP on these algorithms, a method we introduced in our previous research, to dramatically reduce feature selection time. In addition, we introduced a novel idea

of part-of-speech bias in textual CBR that could affect both RST and conventional approaches. Evaluation of these methodologies with large multi-class and n-ary classification tasks showed that JR, suitably biased, significantly outperforms IG and significantly benefits from RTP. Furthermore, POS bias significantly improved RST feature selection algorithms.

Given that JR significantly outperformed IG on our data, we suspect that Wiratunga *et al*.'s (2004) boosted algorithm, which is based on IG, could significantly benefit from our methodologies. We also conjectured that using an appropriate POS bias could consistently improve IG, and that IG effectively counters bias when the number of cases per class is large. In our future work, we will investigate these conjectures.

Acknowledgements

This research was supported by the Naval Research Laboratory.

References

An, A., Huang, Y., Huang, X., & Cercone, N. (2004). An effective rough set-based method for text classification. *Transactions on Rough Sets*, **2**, 1-13.

Brill, E. (1993). *A corpus-based approach to language learning*. Doctoral dissertation: Department of Computer Science, University of Pennsylvania, Philadelphia, PA.

Bruninghaus, S. & Ashley, K.D. (2003). Combining case-based and model-based reasoning for predicting the outcome of legal cases. *Proceedings of the Fifth International Conference on Case-Based Reasoning* (pp. 65-79). Trondheim, Norway: Springer.

Chouchoulas, A., & Shen, Q. (2001). Rough-set aided keyword reduction for text categorization. *Applied Artificial Intelligence*, **15**, 843-873.

Delany, S.J., Cunningham, P., Doyle D., & Zamolokskikh, A. (2005). Generating estimates of classification confidence for a case-based spam filter. *Proceedings of the Sixth International Conference on Case-Based Reasoning* (pp. 177-190), Chicago, IL: Springer.

Gupta, K.M, & Aha, D.W.(2004). RuMop: A rule-based morphotactic parser. *Proceedings of the International Conference on Natural Language Processing* (pp. 280-284). Hyderabad, India: Allied Publishers.

Gupta, K.M., Moore, P.G., Aha, D.W., & Pal, S.K. (2005). Rough set feature selection methods for case-based categorization of text documents. *Proceedings of the First International Conference on Pattern Recognition and Machine Intelligence* (pp. 792-798). Kolkata, India: Springer.

Johnson, D.S. (1974). Approximation algorithms for combinatorial problems, *Journal of Computer and System Sciences*, **9**, 256-278.

Lang, K. (2006). 20 News group dataset. [http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/news20.html]

Li, Y., Shiu, S.C.K., & Pal, S. (2006). Combining Feature Reduction and Case Selection in Building CBR Classifiers. In D.W. Aha, K.M. Gupta, & S.K. Pal (Eds.) *Case-Based Reasoning and Data Mining.* Hoboken, NJ: John Wiley & Sons.

Montazemi, A.R. & Gupta, K.M. (1997). A framework for retrieval in case-based reasoning systems. *Annals of operations research*, **72**, 51-73.

Pal, S.K., & Shiu, S.C.K. (2004). *Foundations of soft case-based reasoning*. Hoboken, NJ: Wiley.

Pawlak, Z. (1991). *Rough sets*. Norwell, MA: Kluwer Academic Publishers.

Quirk, R., Greenbaum, S., Leech, G., & Svartvik, J. (1985). *A comprehensive grammar of the English language*. New York, NY: Longman.

Reuters (2006). Reuters-21578 Evaluation Data. Retrieved on April, 12, 2005 from [http://www.daviddlewis.com/resources/testcollections/reuters21578/]

Skowron, A., (1995). Extracting laws from decision tables. *Computational Intelligence*, **11**(2), 371-388.

Weber, R.O., Ashley, K.D., & Brüninghaus, S. (2005). Textual case-based reasoning. To appear in *Knowledge Engineering Review*, **20**(3).

Wilson, D.C., & Bradshaw, S. (2000). CBR textuality. *Expert Update*, **3**(1), 28-37.

Wiratunga, N., Koychev, I., & Massie, S. (2004). Feature selection and generalization for retrieval of textual cases. *Proceedings of the Seventh European Conference on Case-Based Reasoning* (pp. 806-820). Madrid, Spain: Springer.

Yang, Y., & Pederson, J. (1997). A comparative study of feature selection in text categorization. *Proceedings of the Fourteenth International Conference on Machine Learning* (pp. 412-420). Nashville, TN: Morgan Kaufmann.